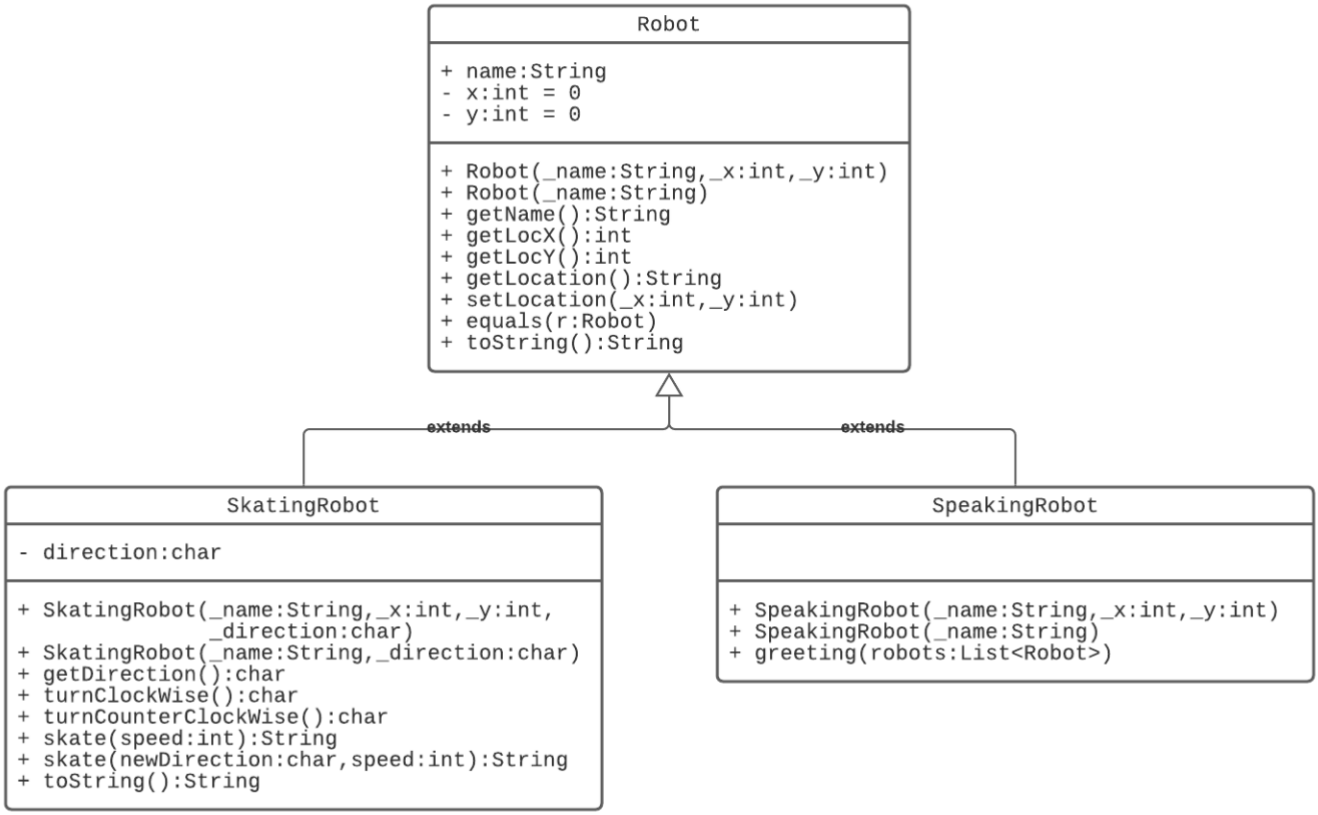
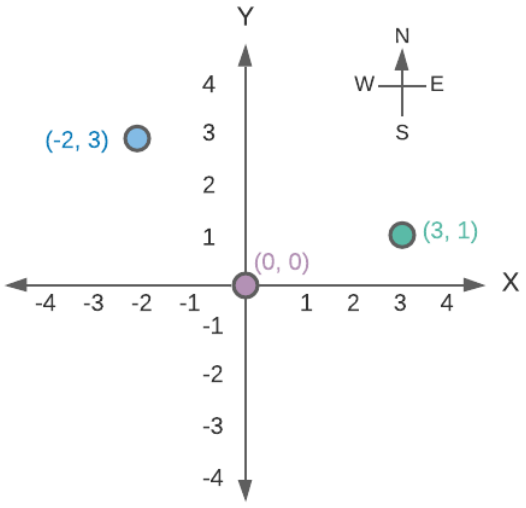
**[Question X ]**

Write a program to control robots. You can refer to the UML class diagram below for more details about attributes, constructors and methods.




The general robot has name and location X and Y in 2-Dimentional space as shown in the picture on the right. The default value is (0, 0). The Robot class is provided in **Robot.java (DO NOT modify)**. You will have to create two special types of robot.




1) The `SkatingRobot` can turn around and skate with a speed in the direction of north(N), east(E), south(S) or west(W). The skeleton of this class is already provided in **SkatingRobot.java**.

2) The `SpeakingRobot` can speak to other robots, who are at the same location. You will have to create a new class (**SpeakingRobot.java**) from scratch.

To test your program, you can execute the main method in **RobotTester.java** file and compare your output with the expected output OR you can run Junit5 testcase using **Tester.java.**

**Task1 (30 points):**

Implement the `SkatingRobot` class. You can find more explanation of all methods in the java file (SkatingRobot.java)

- Two (overloaded) constructor methods.
- `getDirection()`: to return the current direction of the robot
- `turnClockWise()` and `turnCounterClockWise()`: to turn the direction of the robot clockwise and counter clockwise accordingly
- `skate(int speed)`: to skate to a new location with the given speed. The method will throw an `IllegalArgumentException` if the given speed is invalid.
- `skate(char newDirection, int speed)`: To turn the robot into a new direction and skate to a new location with the given speed. The method will throw an `IllegalArgumentException` if the given direction is invalid.
- `toString()`: to return information about this robot as shown in the expected output

**Expected Output**

```
--------- Testing SkatingRobot ---------
Newbie is at location (2, 5) and facing at direction N
* Turns clockwise -> E
* Skates with speed 5 -> (7, 5)
* Turns to direction E and skats with speed 5 -> E (9, 5)
Newbie is at location (9, 5) and facing at direction E

Testing invalid arguments!
** Skates with speed 0
Invalid speed (must be positive number)
** Turns to direction D and skaptes with speed 8
Invalid direction (N, S, E, and W only)
Newbie is at location (9, 5) and facing at direction E
```

**Task 2 (10 points):**

Create a new file named **SpeakingRobot.java** to implement the `SpeakingRobot` class. This class contains following methods

- two (overloaded) constructor methods with the same behavior as the general Robot
- `greeting(List<Robot> robot)`: that takes a list of robots and print the message to greet only the robot who are at the same location. The greeting message must contain other robots' name and the same information as shown in the expected output.

**Expected Output**

```
--------- Testing SpeakingRobot ---------
*** Greetings other robots at the same location.
Hi [Python, Go, React], I'm Java. Nice to meet you.
*** Moves to a new location (10, 1) and greetings again.
Hi [Dart, Ruby], I'm Java. Nice to meet you.
```

## **Files to submitted**: SkatingRobot.java, and SpeakingRobot.java